APPLICATION FOR UNITED STATES PATENT

For

APPARATUS AND METHOD FOR AAL2 PACKET SWITCHING ON AN ATM SWITCH CORE

Inventor:

Safa Almalki

Prepared by:

BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLP
32400 Wilshire Boulevard
Los Angeles, CA 90025-1026
(408) 720-8598

Attorney's Docket No.: 81862P212

"Express Mail" mailing label number: EL617210733US
Date of Deposit: June 20, 2001
I hereby certify that I am causing this paper or fee to be deposited with the United States Postal Service "Express Mail Post Office to Addressee" service on the date indicated above and that this paper or fee has been addressed to the Commissioner for Patents, Washington, D. C. 20231

Leah Resendez
(Typed or printed name of person mailing paper or fee)

Leah Resendez
(Signature of person mailing paper or fee)

June 20, 2001
(Date signed)

APPARATUS AND METHOD FOR AAL2 PACKET SWITCHING ON AN ATM SWITCH CORE

FIELD OF THE INVENTION

[0001] The field of invention relates generally to networking; and, more specifically, to an apparatus and method for AAL2 packet switching on an ATM switch core.

BACKGROUND

[0002] AAL2 is a packet based ATM standard (ITU-T I.366) that allows multiple users to share the same Virtual Path Identification / Virtual Channel Information (VPI/VCI). Figure 1a elaborates on the AAL2 approach in more detail. Within an AAL2 framework, AAL2 packets (such as the packet formed by header 101a and payload 102a and the packet formed by header 101b and payload 102b) are transported over an ATM network.

[0003] As ATM cells are used to transport information over an ATM network, the AAL2 packets observed in Figure 1a are "broken down" into separate pieces of information that are each carried by individual ATM cells (which are not shown in Figure 1a for simplicity). Upon their reception, the payload of the ATM cells are pieced back together to form the flow of packets.

[0004] Frequently, neighboring packets are associated with different connections. For example, the packet formed by header 101a and payload 102a may be destined for a first user while the packet formed by header 101b and payload 102b may be destined for a second user. Thus, within each header 101a, 101b, is a Connection Identification

(CID) label that informs a receiving node as to which connection the corresponding payload 102a, 102b belongs.

[0005] Under the current ITU-T I.366 specification, 6 bits are reserved within the CID which allows for 248 different connections. As such, up to 248 different AAL2 connections can be carried by a single VPI/VCI ATM connection. Note that information within different AAL2 packets may be carried by the same ATM cell. That is, the payload of a single ATM cell may contain information at (and on either side of) the transition X from the first packet to the second packet seen in Figure 1a. As such, a pointer may also be included in the packet header to signify where a packet starts and where a packet ends.

[0006] Figure 1b shows a model 103 for an AAL2 switch. Ingress ATM cells are received on a plurality of ingress lines (such as ingress line 108). A line, such as an ingress line, may be any line that carries ATM cells (such as any OC-n line or STS-n line). The ATM cell traffic from the ingress lines are collected on the ingress portion 105a, 106a, 107a of one or more line cards.

[0007] The ingress portion 105a, 106a, 107a of each of the line cards reconstructs AAL2 packets from the payloads of the received ATM cells. The AAL2 packets are then forwarded to an AAL2 switch core 104 that switches each ingress AAL2 packet, based upon its respective CID, to the appropriate egress line card portion 105b, 106b, 107b. The AAL2 switch core 104 can change the CID label to an appropriate egress value for transmission from the switch 103. Note that, frequently, a single line card has both an ingress portion and an egress portion.

[0008] Thus, for example, ingress portion 105a and egress portion 105b may be viewed as separate portions of a first line card; ingress portion 106a and egress portion 106b may be viewed as separate portions of a second line card; and ingress portion 107a and egress portion 107b may be viewed as separate portions of a third line card. Upon receipt of an AAL2 packet at an egress portion, the AAL2 packet may be broken down into pieces that are carried by different ATM cells which are subsequently transmitted along the appropriate egress line (such as egress line 109).

[0009] A problem, however, is the loss of economies of scale when implementing an AAL2 switch with an AAL2 switch core 104. That is, AAL2 is a specific type of networking approach. As other types of networking approaches are in common usage (e.g., pure ATM, AAL3/4, AAL5, Packets Over SONET (PoS), etc.), a switch having an AAL2 switch core 104 can not efficiently switch the traffic associated with these other types of networking approaches.

SUMMARY

[0010] An apparatus is described having an ATM switch core, wherein the ATM switch core has an input coupled to an ingress AAL2 switch engine and an output coupled to an egress AAL2 switch engine. The ingress AAL2 switch engine has a first look-up table that can store: 1) an identification label for an AAL2 virtual channel through said ATM switch core; and, 2) an AAL2 egress connection identification label for an ingress AAL2 packet to be carried by the AAL2 virtual channel. The egress AAL2 switch engine has a second look-up table that can store an egress CID and egress VPI/VCI for the AAL2 egress connection identification label.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011] The present invention is illustrated by way of example, and not limitation, in the Figures of the accompanying drawings in which:

[0012] **Figure 1a** shows a sequence of AAL2 packets;

[0013] **Figure 1b** shows a model for an AAL2 switch;

[0014] **Figure 2** shows an embodiment of an architecture that implements the functionality of an AAL2 switch with an ATM switch core;

[0015] **Figure 3** shows a look up table that may be used by an ingress AAL2 switch engine of **Figure 2**;

[0016] **Figure 4** shows a look up table that may be used by an egress AAL2 switch engine of **Figure 2**; and

[0017] **Figure 5** shows a method for implementing AAL2 switching on an ATM switch core.

DETAILED DESCRIPTION

[0018] An apparatus is described having an ATM switch core, wherein the ATM switch core has an input coupled to an ingress AAL2 switch engine and an output coupled to an egress AAL2 switch engine. The ingress AAL2 switch engine has a first look-up table that can store: 1) an identification label for an AAL2 virtual channel through said ATM switch core; and, 2) an AAL2 egress connection identification label for an ingress AAL2 packet to be carried by the AAL2 virtual channel. The egress AAL2 switch engine has a second look-up table that can store an egress CID and egress VPI/VCI for the AAL2 egress connection identification label.

[0019] A solution to the economy of scale problem presented in the background is to implement the functionality of an AAL2 switch upon an ATM switch core. An ATM switch core switches ATM cells based upon their VPI/VCI information. Various types of ATM switch core architectures exist, such as crossbar switches and shared output buffered switches (to name just a few). As ATM cells are a fundamental transport unit for various types of networking approaches (e.g., pure ATM, AAL2, AAL3/4, AAL5, PoS, etc.), an ATM switch core is better suited to cost effectively manage the switching function of these various approaches.

[0020] Figure 2 shows an embodiment of an architecture that implements the functionality of an AAL2 switch with an ATM switch core. Ingress ATM cells are received on a plurality of ingress lines (such as ingress lines 213_1, 213_2, through 213_x). The ATM cell traffic from the ingress lines are collected on the ingress portion 201_1, 201_2, . . . 201_n of one or more line cards.

[0021] Each ingress line may be deemed to correspond to an ingress port of the switch architecture 200 of Figure 2. In the embodiment of Figure 2, the ATM Rx units (e.g., ATM Rx unit 203) provide ATM cells that are received from a plurality of ingress ports. For example, each ATM Rx unit may correspond to an ATM physical layer (PHY) interface device. Those ATM cells that are not associated with an AAL2 packet flow (as identified by their VPI/VCI address) are forwarded to the ATM switch core 204 (e.g., along logical path 205 for cells provided by ATM Rx unit 203).

[0022] Those ATM cells having a VPI/VCI associated with an AAL2 packet flow are forwarded to an ingress AAL2 switch engine 206. An AAL2 ingress switch engine 206, as described in the exemplary details that follow, is a functional unit that effectively converts information that characterizes an ingress AAL2 packet (e.g., its CID value, its VPI/VCI address, its ingress port, etc.) into: 1) information that is used by the ATM switch core 204 to direct the AAL2 packet to its appropriate egress line card portion; and 2) information that is used by the appropriate egress line card portion to understand which egress port, egress CID value and egress VPI/VCI information should be implemented into the outgoing AAL2 packet and the outgoing ATM cells used to transport it.

[0023] In the embodiment of Figure 2, the AAL2 switch engine 206 reassembles the payloads from the ATM cells it receives. Note that those ATM cells having the same VPI/VCI can correspond to a multi-user stream of AAL2 packets such as the stream of packets shown back in Figure 1a. As a single ingress port may support multiple VPI/VCI addresses that each support an AAL2 packet stream; and, as multiple ingress

ports may be received by the ATM Rx unit 203, a plurality of AAL2 packet streams may be processed by the ingress AAL2 switch engine 206 of Figure 2.

[0024] The switch engine 206 effectively directs each AAL2 packet, regardless as to which AAL2 packet stream it arrived on, to its appropriate egress line card portion (e.g., egress line portion 202_1, 202_2, ... or 202_n as seen in Figure 2) through the ATM switch core 204. Figure 3 shows an exemplary look-up table 300 that may be employed by an ingress AAL2 switch engine 206 to help direct a particular AAL2 packet to its appropriate egress line card portion.

[0025] The inputs to the look-up table 300 of Figure 3 include: 1) the CID of the AAL2 packet to be directed; 2) the VPI/VCI address of the AAL2 packet stream that the AAL2 packet to be directed arrived with; and 3) the port on which the AAL2 packet arrived. The combination of these three items are unique to any AAL2 packet arriving at the ATM Rx unit 203 for a particular user. That is, for example, CID values may be reused for different connections across the various VPI/VCI address values that correspond to an AAL2 packet stream. In the embodiment of Figure 3, VPI/VCI values may conceivably also be reused for different AAL2 streams across the various input ports that are received by the ATM Rx unit 203.

[0026] Thus, in the embodiment of Figure 3, each input/output entry of the look up table 300 corresponds to a particular AAL2 connection supported by the switch 200 of Figure 2. As seen in the embodiment of Figure 3, the output contents of the look up table 300 provide: 1) The pipe number (P#); and 2) the egress identification number (EID). The P# is information that is used by the ATM switch core 204 to direct the AAL2 packet to its appropriate egress line card portion.

[0027] The EID information (which is appended to the payload of the AAL2 packet as it traverses the ATM switch core 204) is transported with the payload of a received AAL2 packet as it traverses the ATM switch core 206. The EID information helps the appropriate egress line card portion understand which egress port, egress CID value and egress VPI/VCI information should be implemented into the outgoing AAL2 packet and the outgoing ATM cells used to transport it. In the following, a discussion of the P# will precede a discussion of the EID.

[0028] In the embodiment of Figures 2 and 3, the P# effectively points to (or otherwise triggers the use of) a local, virtual channel within the ATM switch core 204 used to carry AAL2 packet flows. Exemplary AAL2 virtual channels 208_1, 208_2, 208n that respectively couple ingress line card portion 201n to each egress line card portion 202_1, 202_2 ...202_n are shown in Figure 2. The AAL2 virtual channels 208_1, 208_2, 208_n may be established within the ATM switch core 204 as permanent connections in light of the AAL2 packet flows that the switch 200 is configured to support.

[0029] For example in one approach, if an ingress line card portion is configured to actively receive AAL2 packet flows (e.g., by enabling the ingress switch AAL2 engine), an AAL2 virtual channel is established through the ATM switch core 204 from the ingress line card to each egress line card portion 202_1, 202_2 through 202_n. For example, as seen in Figure 2, once ingress line card portion 201_n is configured to actively receive an AAL2 packet flow, AAL2 virtual channels 208_1, 208_2, 208_n are created within the ATM switch core 204 in response. Note that from the perspective of the ATM switch core 204, each AAL2 virtual channel 208_1, 208_2, 208_n may be viewed as an ATM virtual channel.

[0030] This provides the newly configured AAL2 service with the versatility to switch AAL2 packets to any egress destination. In an alternate embodiment, the egress line of a particular AAL2 packet connection is fixed. As such, only one virtual channel that connects the ingress line card portion to the appropriate egress line card portion is needed.

[0031] In a further embodiment of either of these approaches, the capacity of the virtual channel through the ATM switch core 204 is extended to support more than one AAL2 packet flow (e.g., up to 248 different AAL2 connections). This allows an ingress line card portion that manages the reception of different AAL2 packet connections (e.g., if ingress line card portion 201_n is configured to manage different AAL2 connections received across ingress lines 213_1 through 213_n) to "pipe" AAL2 packets having the same egress line card portion destination (e.g., egress line card portion 202_1) over a common virtual channel through the ATM switch core (e.g., virtual channel 208_1). For example, under current implementations, each VPI/VCI recognized by the ATM switch core 204 can have as many as 248 connections via the CID header parameter.

[0032] As discussed, non AAL2 packet flows (such as pure ATM cell switching flows) avoid the ingress AAL2 switch engine 206 and are presented to the ATM switch core 204 and switched by normal means. For example, for pure ATM flows on ingress line card portion 201_n, ATM cells traverse logical path 205 and are presented to the ATM switch core 204. The ATM switch core 204 may switch these cells based upon their VPI/VCI information.

[0033] AAL2 packets are also effectively presented to the ATM switch core 204 after processing by the ingress AAL2 switch engine 206. In an embodiment, AAL2

packet payloads destined for the same egress line card portion are inserted into one or more ATM cells having a VPI/VCI address that corresponds to the appropriate virtual channel (e.g., AAL2 virtual channel 208_1 for egress line card portion 202_1, AAL2 virtual channel 208_2 for egress line card portion 202_2, and AAL2 virtual channel 208_n for egress line card portion 202_n). In this approach, this VPI/VCI address corresponds to the P# that is listed in the ingress AAL2 switching engine look up table 300 shown in Figure 3.

[0034] As such, the P# effectively points to the appropriate AAL2 virtual channel and also provides a means (e.g., a VPI/VCI address) that is recognized by an ATM switch core 204 to appropriately direct traffic through the switch core 204. Note that the stream of ATM cells carrying AAL2 packet payload information across the same virtual channel (through the ATM switch core 204) may be formatted with appropriate AAL2 overhead. As such, a stream of AAL2 packets (similar to that shown in Figure 1a) are reassembled from the cell payloads upon their reception at the destination egress line card portion.

[0035] In this approach, the cells used to carry the AAL2 packet stream are formatted with the VPI/VCI found as the P# within look up table 300 of Figure 3. In this case, the egress AAL2 switch engine (e.g., egress AAL2 switch engine 210 for destination line card portion 202_1) performs the appropriate AAL2 packet reassembly. Note that, upon the reception of ATM cells from the ATM switch core 204 at an egress line card portion, those ATM cells having a VPI/VCI address that corresponds to an AAL2 virtual channel through the ATM switch core 204 are diverted for processing by the egress AAL2 switching engine 210. That is, as virtual channels carry AAL2

information, the egress AAL2 switching engine 210 is invoked for cells carried along these channels.

[0036] Those ATM cells not having a VPI/VCI address that corresponds to an AAL2 virtual channel through the ATM switch core 204 follow logical path 211 to the ATM Tx unit 212. Note that, these "non AAL2" cells may have their VPI/VCI address changed by the ATM switch core 204 consistent with ATM cell switching technology. For those cells having a VPI/VCI address that corresponds to an AAL2 virtual channel through the ATM switch core 204, however, no such change is necessary as the virtual channel may be viewed as local to the ATM switch core 204.

[0037] Recall that the EID information found in the look-up table 300 of Figure is effectively appended to the payload of an AAL2 packet as it traverses the ATM switch core 204. The EID, as described in more detail immediately below, helps the appropriate egress AAL2 switch engine 210 understand which egress port, egress CID value and egress VPI/VCI information should be implemented into the outgoing AAL2 packets and the outgoing ATM cells used to transport them.

[0038] The EID may be viewed as a local extension to the CID. That is, recall that the CID identifies AAL2 connections that are particular to a VPI/VCI. In the embodiment of Figure 2, the EID identifies AAL2 connections that are particular to an egress line card portion. That is, each egress line card portion 202_1, 202_2, ...202_n "keeps track of" separate EID numbers that are used to identify each AAL2 connection that it controls the transmission of.

[0039] Thus, for example, when updating the look up table 300 for an ingress switch engine (e.g., ingress switch engine 206) with information for a new connection,

the particular EID number given to the connection is determined in light of the available EID numbers that may be used by the particular egress switch engine (e.g., egress switch engine 210) that will handle the traffic for the connection. In order for the switch to handle more than one VPI/VCI address worth of AAL2 traffic per egress line card portion (i.e., in order for the switch to handle more than 248 outgoing AAL2 connections per egress line card portion), the size of the EID field should be larger than the size of the CID field.

[0040] In an embodiment, the EID field size is 16 bits which allows for over 64,000 AAL2 connections per egress line card portion. Alternate embodiments may reduce the EID field size to less than 16 bits (if less than 64,000 AAL2 connections per egress line card portion is desired) or may expand the EID field size to more than 16 bits (if more than 64,000 AAL2 connections per egress line card portion is desired). Various techniques may be used to append the EID number to an AAL2 packet prior to processing by the ATM switch core. For example, in an embodiment where a stream of AAL2 packets (similar to Figure 1a) are segmented and carried by ATM cells 204 over the ATM switch core, the EID may be comprised of the CID field plus additional header space within the AAL2 packet (e.g., reserved bit locations).

[0041] Alternatively or in combination, the ATM switch core interface 207 may allow for a control header that is presented along with each cell offered to the ATM switch core 204 for service. Such a control header may be formatted to include the EID number particular to the AAL2 packet(s) being carried by the ATM cell. Such a control header may also be used to identify an AAL2 virtual channel number (i.e., a P# other than a VPI/VCI address) that is recognized by the ATM switch core 204.

[0042] After the payload for an AAL2 packet and its corresponding EID number is recognized by an egress AAL2 switch engine 210, the egress AAL2 switch engine 210 looks up appropriate information for the outgoing connection based upon the EID number. Figure 4 shows an embodiment of such a look up table 400. As mentioned just above, the look up is based upon the EID number. As such, the EID number is the look up table 400 input parameter.

[0043] The output parameters in the look up table embodiment 400 of Figure 4 include: 1) the CID number for the outgoing connection; 2) the VPI/VCI address to be used by the ATM cells that will be used to transport the outgoing AAL2 packet; and 3) the outgoing port (e.g., outgoing network line such as any of lines 214_1, 214_2, through 214_y). Note that, consistent with the operation of an AAL2 switch, CID numbers are switched from ingress to egress as discussed with respect to Figure 1b. Furthermore, as is clear from above, an egress AAL2 switch engine effectively converts the EID into information that is used to properly transmit the AAL2 packet (e.g., its CID, its corresponding VPI/VCI and its corresponding output port).

[0044] Once the information is gathered for the outgoing connection from the look-up table, the proper AAL2 header structure is formed (including the CID found in the table 400) and appended to the payload. The AAL2 header and payload is then segmented as necessary into ATM cells having the VPI/VCI found in the table. These cells are then forwarded to the ATM Tx unit (such as ATM Tx unit 212) with the outgoing port information found in the look up table so that the cells are transported over the correct outgoing line. The ATM Tx unit 212 may correspond to an ATM transmission PHY device.

[0045] Figure 5 shows a methodology that reviews the AAL2 switching sequence approach discussed above. Note that, although the methodology of Figure 5 is written to correspond to Common Part Sublayer (CPS) AAL2 packets, other types of AAL2 packet characterizations (such as Service Specific Convergence Sublayer (SSCS)) also apply to the methodology of Figure 5.

[0046] Upon the reception of ATM cells that carry AAL2 packet information, the ingress AAL2 packet payload is tagged 501 with an egress identification label (e.g., the EID discussed above). The ingress AAL2 packet payload and egress identification label are then formatted 502 for switching by an ATM switch core (e.g., by incorporating the payload as part of an AAL2 packet stream (that includes the EID in its header space) which is further segmented or otherwise incorporated into a series of ATM cells).

[0047] The ingress AAL2 packet payload and its corresponding EID are then switched 503 by the ATM core to the appropriate egress line card portion. The EID is then used at the egress line card to identify 504 the CID and VPI/VCI information for the outgoing connection the AAL2 payload is to be delivered over. Note that the ingress or egress AAL2 switch engines may be implemented as a processor that performs the methodologies described above in software, as dedicated logic that performs the methodologies described above, or as some combination of both.

[0048] Thus, embodiments of the present description may be implemented not only within a semiconductor chip but also within machine readable media. For example, the designs discussed above may be stored upon and/or embedded within machine readable media associated with a design tool used for designing semiconductor devices. Examples include a netlist formatted in the VHSIC Hardware Description

Language (VHDL) language, Verilog language or SPICE language. Some netlist examples include: a behaviorial level netlist, a register transfer level (RTL) netlist, a gate level netlist and a transistor level netlist. Machine readable media also include media having layout information such as a GDS-II file. Furthermore, netlist files or other machine readable media for semiconductor chip design may be used in a simulation environment to perform the methods of the teachings described above.

[0049] Thus, it is also to be understood that embodiments of this invention may be used as or to support a software program executed upon some form of processing core (such as the CPU of a computer) or otherwise implemented or realized upon or within a machine readable medium. A machine readable medium includes any mechanism for storing or transmitting information in a form readable by a machine (e.g., a computer). For example, a machine readable medium includes read only memory (ROM); random access memory (RAM); magnetic disk storage media; optical storage media; flash memory devices; electrical, optical, acoustical or other form of propagated signals (e.g., carrier waves, infrared signals, digital signals, etc.); etc.

[0050] In the foregoing specification, the invention has been described with reference to specific exemplary embodiments thereof. It will, however, be evident that various modifications and changes may be made thereto without departing from the broader spirit and scope of the invention as set forth in the appended claims. The specification and drawings are, accordingly, to be regarded in an illustrative rather than a restrictive sense.